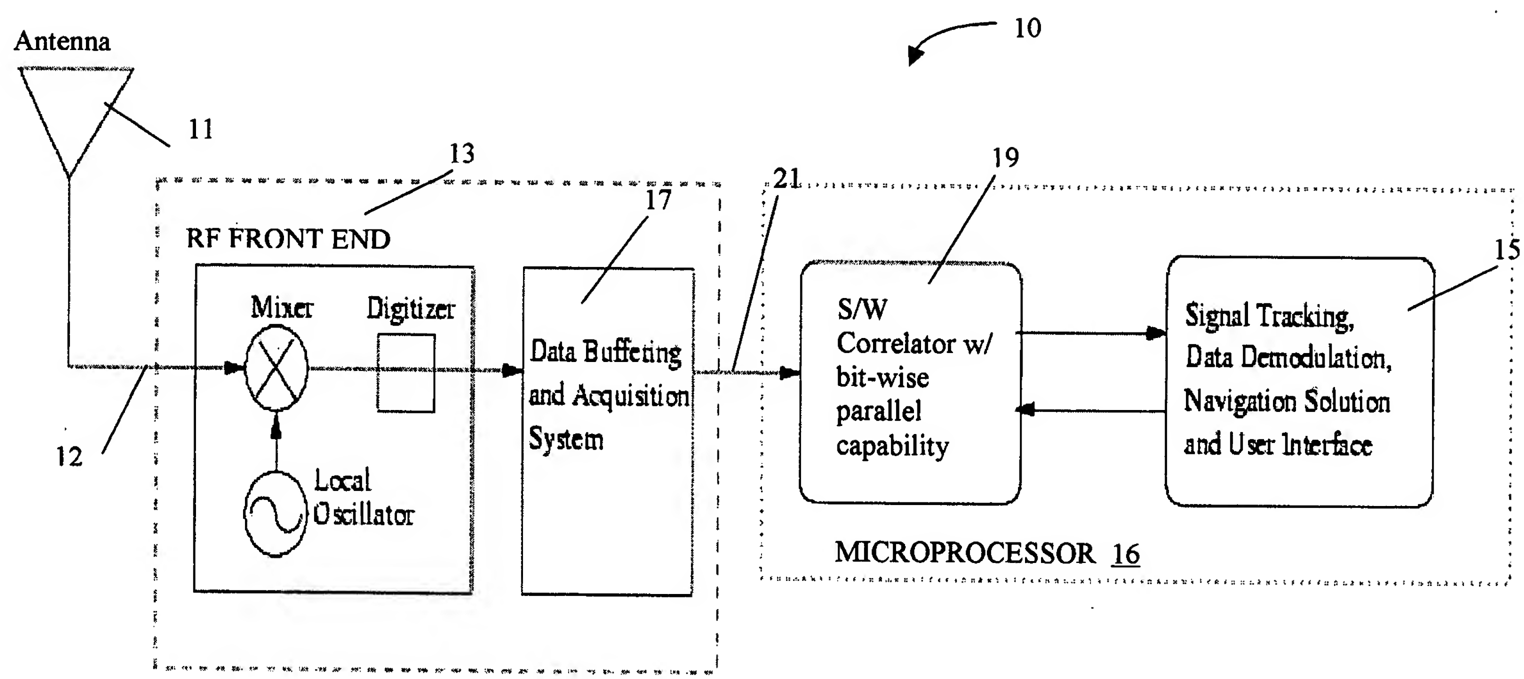

10
Antenna
11
12
13
RF FRONT END
Mixer
Digitizer
Local
Oscillator
17
Data Buffeting
and Acquisition
System
21
19
S/W
Correlator w/
bit-wise
parallel
capability
15
Signal Tracking,
Data Demodulation,
Navigation Solution
and User Interface
MICROPROCESSOR 16

FIG. 1

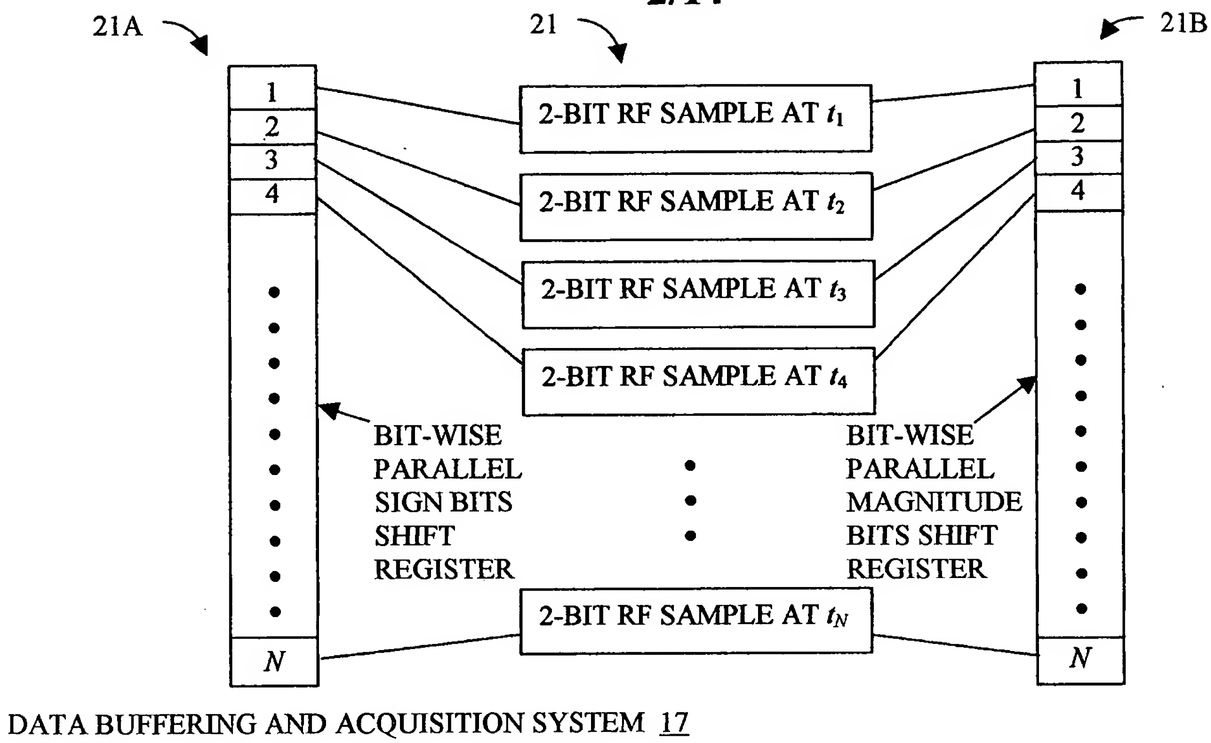

21A
21
21B
1
2
3
4
N
2-BIT RF SAMPLE AT t_1
2-BIT RF SAMPLE AT t_2
2-BIT RF SAMPLE AT t_3
2-BIT RF SAMPLE AT t_4
2-BIT RF SAMPLE AT t_N
BIT-WISE PARALLEL SIGN BITS SHIFT REGISTER
BIT-WISE PARALLEL MAGNITUDE BITS SHIFT REGISTER
DATA BUFFERING AND ACQUISITION SYSTEM 17

FIG. 2A

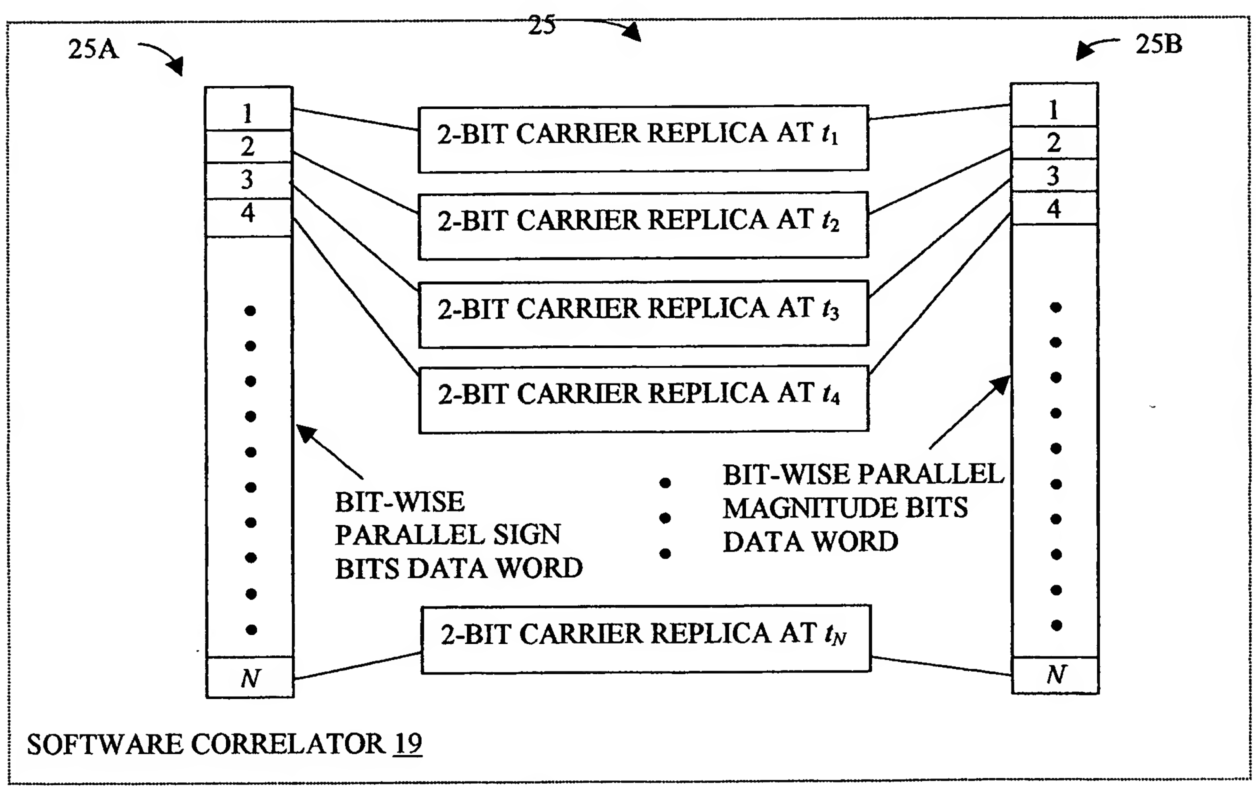

25
25A
25B
1
2
3
4
N
2-BIT CARRIER REPLICA AT t_1
2-BIT CARRIER REPLICA AT t_2
2-BIT CARRIER REPLICA AT t_3
2-BIT CARRIER REPLICA AT t_4
2-BIT CARRIER REPLICA AT t_N
BIT-WISE PARALLEL SIGN BITS DATA WORD
BIT-WISE PARALLEL MAGNITUDE BITS DATA WORD
SOFTWARE CORRELATOR 19

FIG. 2B

Quantity	Sequence								
Sample Times	t_0	t_1	t_2	t_3	t_4	t_5	t_6	t_7	
21 RF Signal	1	1	-1	-1	-1	1	1	1	
Word Representation of Signal	1	1	0	0	0	1	1	1	73
PRN Code replica	1	-1	-1	-1	1	1	1	1	75
Word Representation of PRN Code replica	1	0	0	0	1	1	1	1	
Product of Signal and PRN Code replica	1	-1	1	1	-1	1	1	1	
Word Representation of Product	0	1	0	0	1	0	0	0	

FIG. 2C

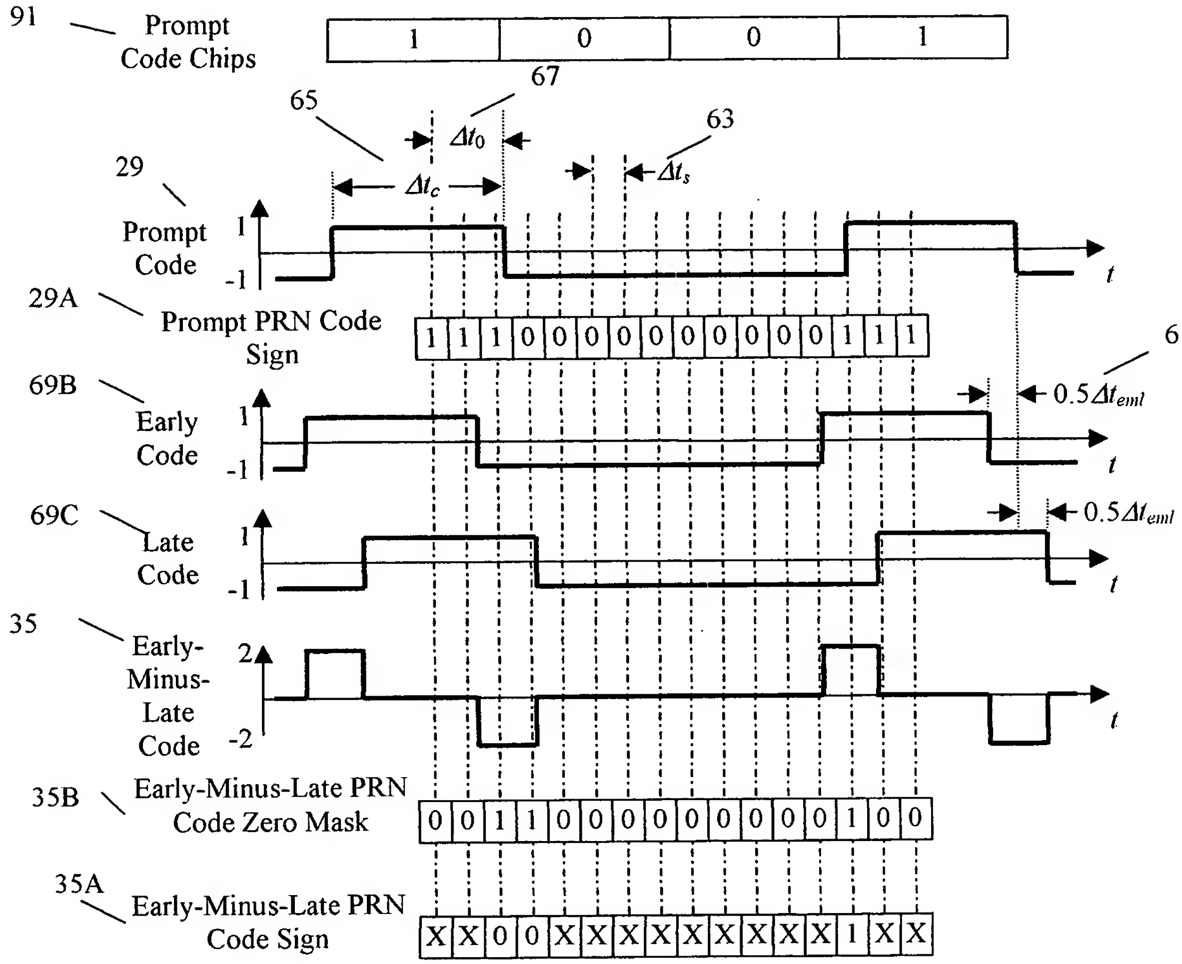

91
Prompt Code Chips
1
0
0
1
67
65
Δt_0
63
Δt_c
Δt_s
29
Prompt Code
1
-1
t
29A
Prompt PRN Code Sign
1 1 1 0 0 0 0 0 0 0 0 0 0 1 1 1
61
69B
Early Code
$0.5\Delta t_{eml}$
69C
Late Code
$0.5\Delta t_{eml}$
35
Early-Minus-Late Code
2
-2
35B
Early-Minus-Late PRN Code Zero Mask
0 0 1 1 0 0 0 0 0 0 0 0 0 1 0 0
35A
Early-Minus-Late PRN Code Sign
X X 0 0 X X X X X X X X X 1 X X

FIG. 2D

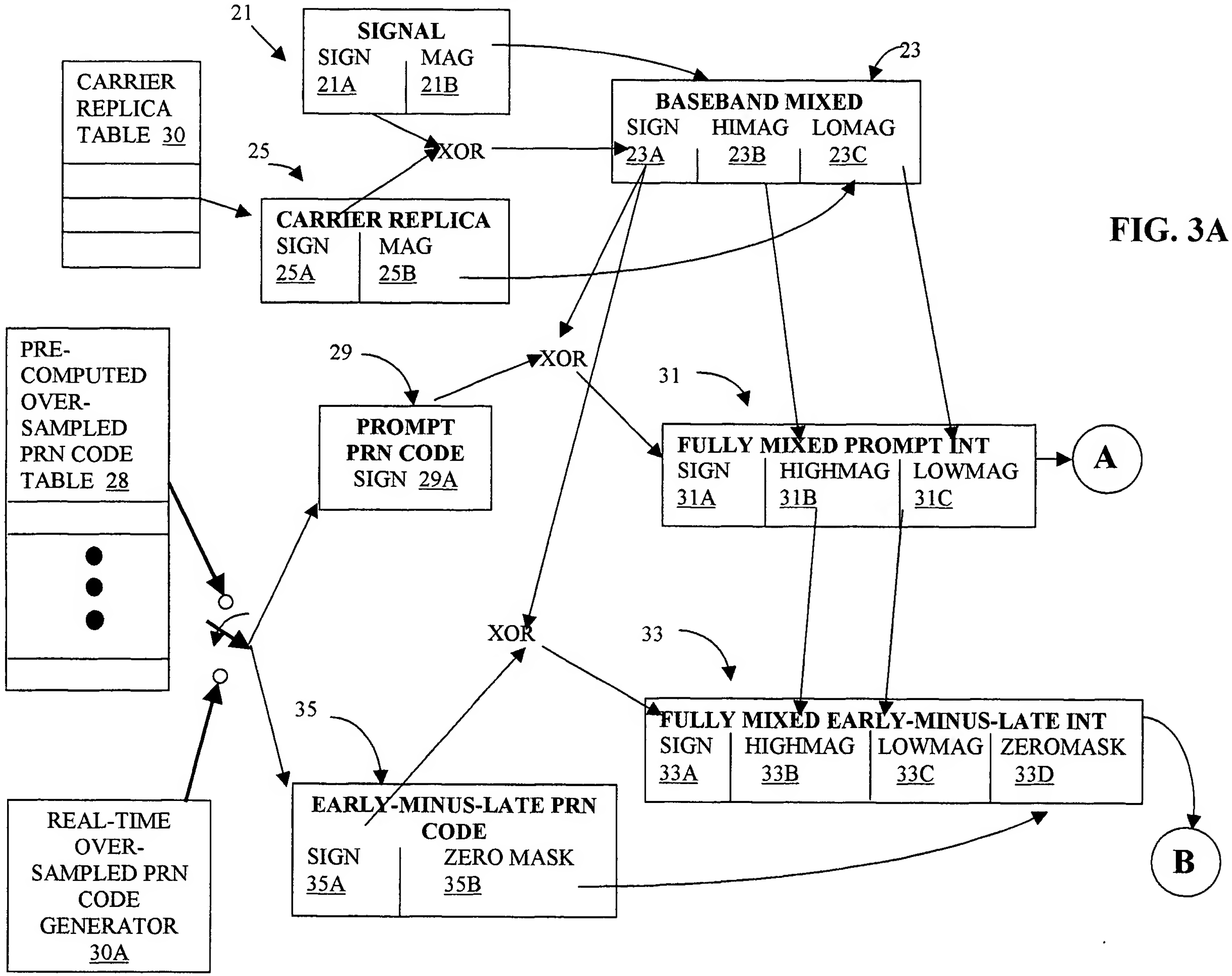

21
SIGNAL
SIGN 21A
MAG 21B
23
BASEBAND MIXED
SIGN 23A
HIMAG 23B
LOMAG 23C
CARRIER REPLICA TABLE 30
25
XOR
CARRIER REPLICA
SIGN 25A
MAG 25B
PRE-COMPUTED OVER-SAMPLED PRN CODE TABLE 28
29
XOR
31
PROMPT PRN CODE
SIGN 29A
FULLY MIXED PROMPT INT
SIGN 31A
HIGHMAG 31B
LOWMAG 31C
A
XOR
33
35
FULLY MIXED EARLY-MINUS-LATE INT
SIGN 33A
HIGHMAG 33B
LOWMAG 33C
ZEROMASK 33D
REAL-TIME OVER-SAMPLED PRN CODE GENERATOR 30A
EARLY-MINUS-LATE PRN CODE
SIGN 35A
ZERO MASK 35B
B

FIG. 3A

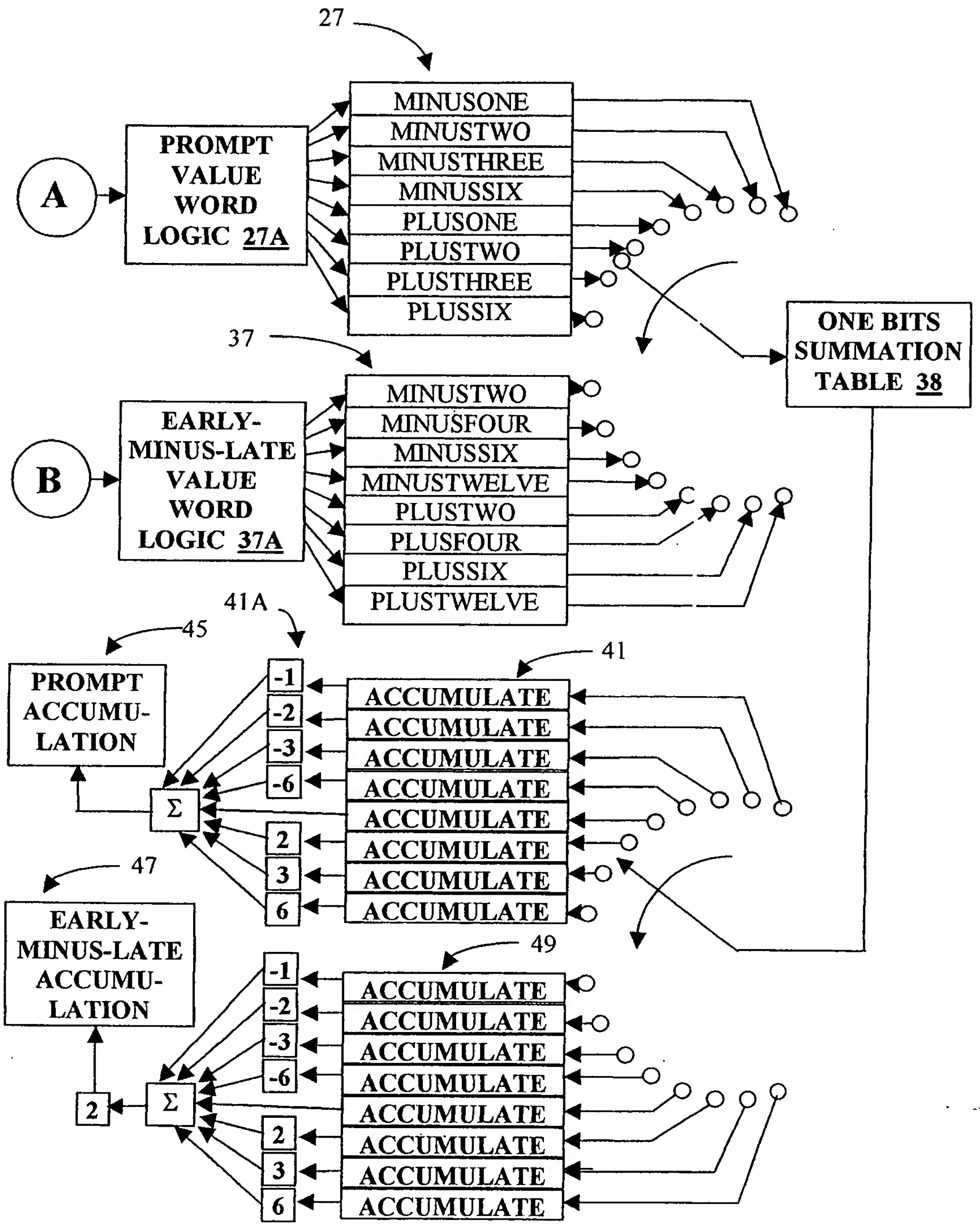

27
A
PROMPT VALUE WORD LOGIC 27A
MINUSONE
MINUSTWO
MINUSTHREE
MINUSSIX
PLUSONE
PLUSTWO
PLUSTHREE
PLUSSIX
ONE BITS SUMMATION TABLE 38
37
B
EARLY-MINUS-LATE VALUE WORD LOGIC 37A
MINUSTWO
MINUSFOUR
MINUSSIX
MINUSTWELVE
PLUSTWO
PLUSFOUR
PLUSSIX
PLUSTWELVE
41A
45
PROMPT ACCUMU-LATION
41
-1
-2
-3
-6
2
3
6
ACCUMULATE
Σ
47
EARLY-MINUS-LATE ACCUMU-LATION
49
2

FIG. 3B

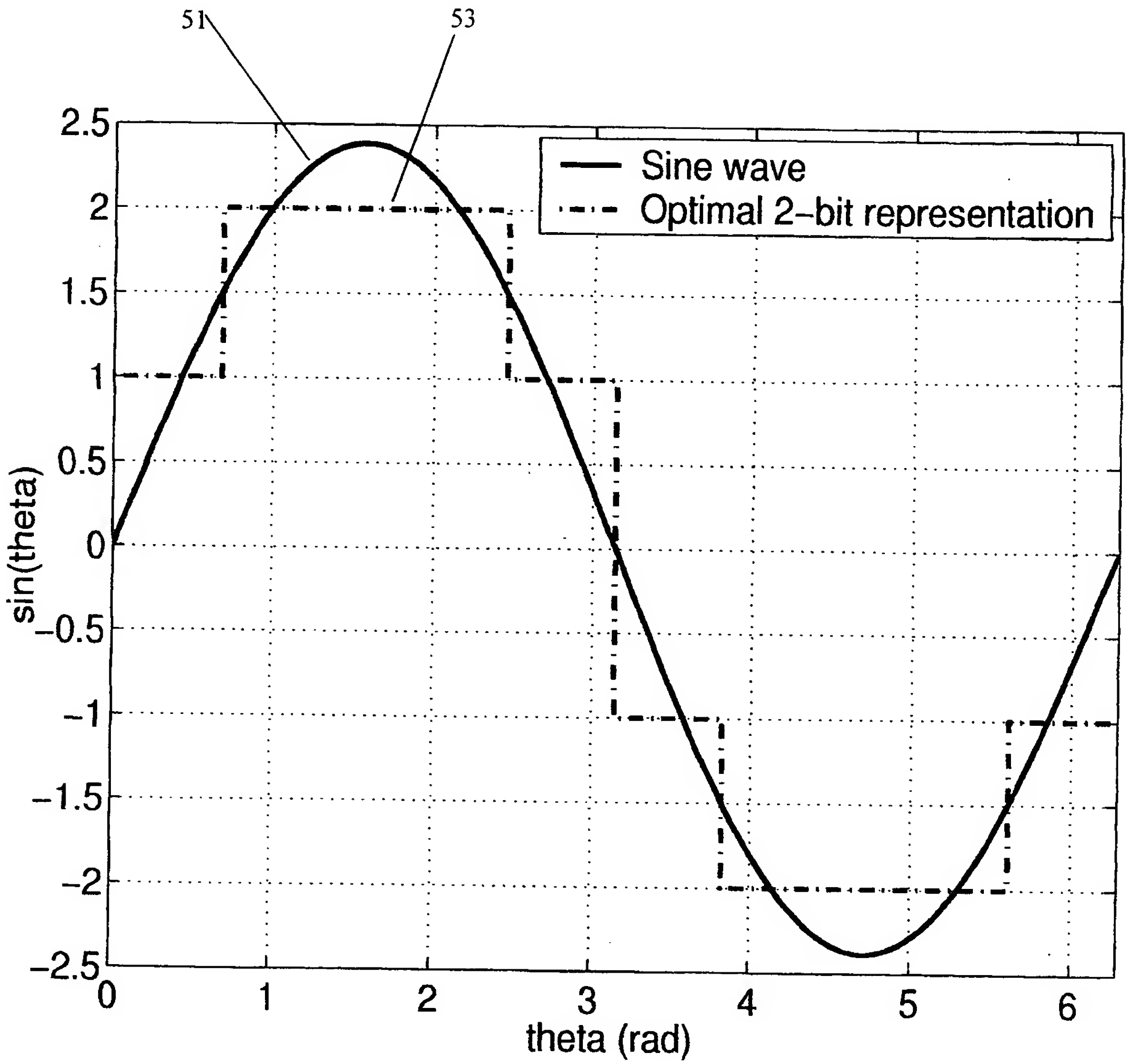


FIG. 3C PRIOR ART

A

REPRESENT SAMPLE SIGNAL DATA 21 FROM AT LEAST ONE CHANNEL AS SIGNAL SIGN 21A AND, IF PRESENT, SIGNAL MAGNITUDE 21B AND SELECT CARRIER REPLICA 25 BASED ON ITS FREQUENCY'S PROXIMITY TO A DESIRED CARRIER REPLICA FREQUENCY, REPRESENT CARRIER REPLICA 25 AS CARRIER REPLICA SIGN 25A AND CARRIER REPLICA MAGNITUDE 25B, 101

MIX SIGNAL DATA 21 TO BASEBAND → COMPUTE BASEBAND MIXED SIGN 23A = XOR (CARRIER REPLICA SIGN 25A, SIGNAL SIGN 21A); COMPUTE BASEBAND MIXED MAGNITUDE 23B/C = f (CARRIER REPLICA MAGNITUDE 25B, SIGNAL MAGNITUDE 21B) 103

SELECT PRN CODE FROM PRN CODE TABLE 28 OR COMPUTE IT USING REAL-TIME OVER-SAMPLED PRN CODE GENERATOR 30A; REPRESENT PROMPT PRN CODE 29 AS PROMPT SIGN 29A; REPRESENT EARLY-MINUS-LATE PRN CODE 35 AS EARLY-MINUS-LATE PRN SIGN 35A AND EARLY-MINUS-LATE PRN ZERO MASK 35B 105

DE-SPREAD BY MIXING IN-PHASE AND QUADRATURE BASEBAND MIXED SIGNALS 23 WITH PROMPT PRN CODE 29 AND EARLY-MINUS-LATE PRN CODE 35: COMPUTE FULLY MIXED PROMPT INTEGRAND SIGN 31A = XOR (BASEBAND MIXED SIGN 23A, PROMPT PRN CODE SIGN 29A); COMPUTE FULLY MIXED PROMPT INTEGRAND MAGNITUDE 31B/C = f (BASEBAND MIXED MAGNITUDE 23B/C); COMPUTE FULLY MIXED EARLY-MINUS-LATE INTEGRAND SIGN 33A = XOR (BASEBAND MIXED SIGN 23A, EARLY-MINUS-LATE PRN CODE SIGN 35A); COMPUTE FULLY MIXED EARLY-MINUS-LATE INTEGRAND HIGH/LOW MAGNITUDE 33B/C= f (BASEBAND MIXED MAGNITUDE 23B/C); COMPUTE FULLY MIXED EARLY-MINUS-LATE ZERO MASK 33D= f (EARLY-MINUS-LATE PRN CODE ZERO MASK 35B); RESULT IS FULLY MIXED IN-PHASE AND QUADRATURE PROMPT INTEGRANDS 31 AND FULLY MIXED EARLY-MINUS-LATE INTEGRANDS 33 107

COMPUTE IN-PHASE AND QUADRATURE PROMPT INTEGRAND VALUE WORDS 27 AND EARLY-MINUS-LATE INTEGRAND VALUE WORDS 37: PROMPT INTEGRAND VALUE WORDS 27 = f (FULLY MIXED PROMPT INTEGRAND SIGN 31A, FULLY MIXED PROMPT INTEGRAND MAGNITUDE 31B/C); COMPUTE EARLY-MINUS-LATE INTEGRAND VALUE WORDS 37 = f (FULLY MIXED EARLY-MINUS-LATE INTEGRAND SIGN 33A, FULLY MIXED EARLY-MINUS-LATE INTEGRAND MAGNITUDE 33B/C, FULLY MIXED EARLY-MINUS-LATE INTEGRAND ZERO MASK 33D) 109

B

FIG. 4A

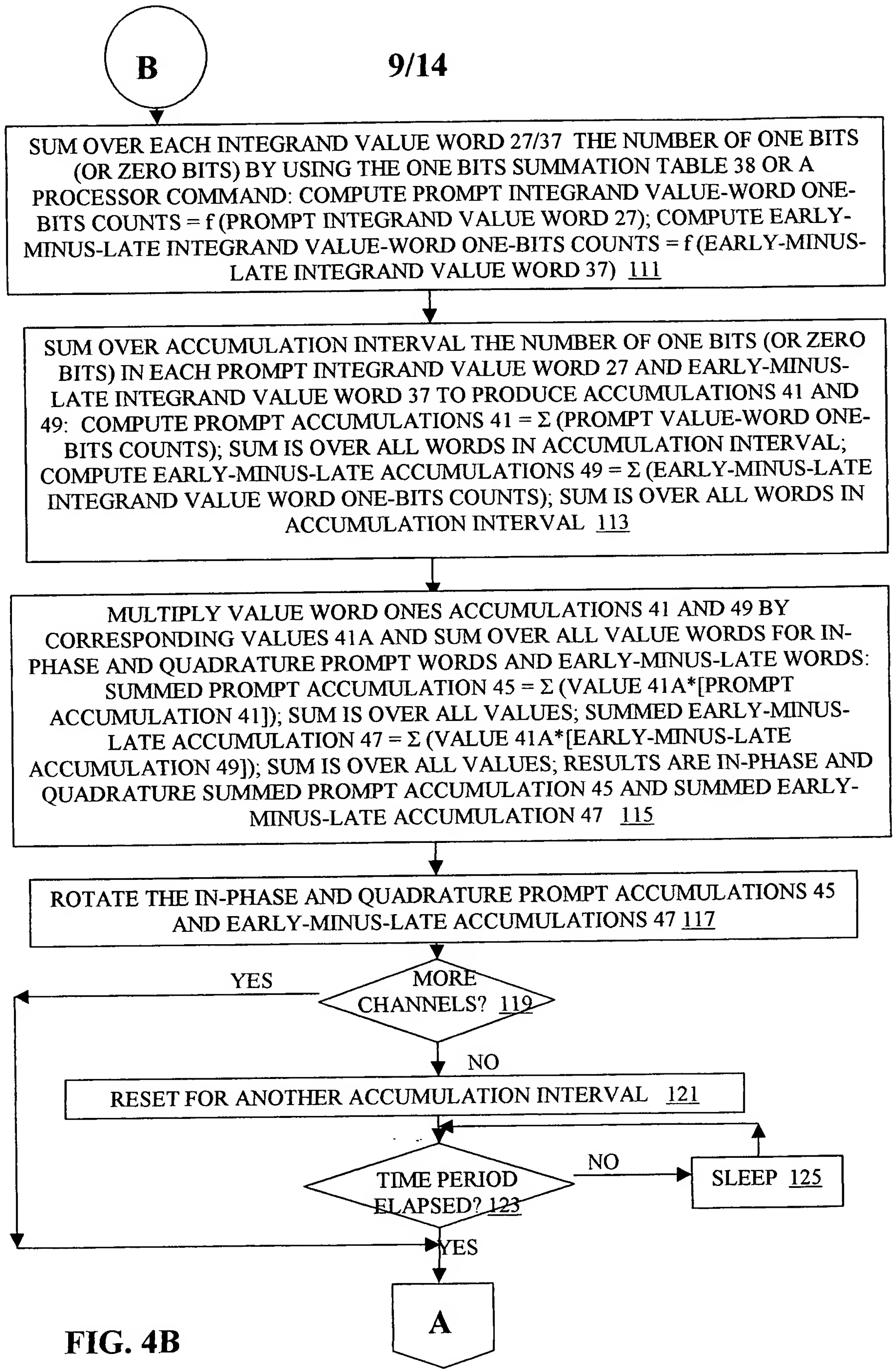

B
SUM OVER EACH INTEGRAND VALUE WORD 27/37 THE NUMBER OF ONE BITS (OR ZERO BITS) BY USING THE ONE BITS SUMMATION TABLE 38 OR A PROCESSOR COMMAND: COMPUTE PROMPT INTEGRAND VALUE-WORD ONE-BITS COUNTS = f (PROMPT INTEGRAND VALUE WORD 27); COMPUTE EARLY-MINUS-LATE INTEGRAND VALUE-WORD ONE-BITS COUNTS = f (EARLY-MINUS-LATE INTEGRAND VALUE WORD 37) 111
SUM OVER ACCUMULATION INTERVAL THE NUMBER OF ONE BITS (OR ZERO BITS) IN EACH PROMPT INTEGRAND VALUE WORD 27 AND EARLY-MINUS-LATE INTEGRAND VALUE WORD 37 TO PRODUCE ACCUMULATIONS 41 AND 49: COMPUTE PROMPT ACCUMULATIONS 41 = Σ (PROMPT VALUE-WORD ONE-BITS COUNTS); SUM IS OVER ALL WORDS IN ACCUMULATION INTERVAL; COMPUTE EARLY-MINUS-LATE ACCUMULATIONS 49 = Σ (EARLY-MINUS-LATE INTEGRAND VALUE WORD ONE-BITS COUNTS); SUM IS OVER ALL WORDS IN ACCUMULATION INTERVAL 113
MULTIPLY VALUE WORD ONES ACCUMULATIONS 41 AND 49 BY CORRESPONDING VALUES 41A AND SUM OVER ALL VALUE WORDS FOR IN-PHASE AND QUADRATURE PROMPT WORDS AND EARLY-MINUS-LATE WORDS: SUMMED PROMPT ACCUMULATION 45 = Σ (VALUE 41A*[PROMPT ACCUMULATION 41]); SUM IS OVER ALL VALUES; SUMMED EARLY-MINUS-LATE ACCUMULATION 47 = Σ (VALUE 41A*[EARLY-MINUS-LATE ACCUMULATION 49]); SUM IS OVER ALL VALUES; RESULTS ARE IN-PHASE AND QUADRATURE SUMMED PROMPT ACCUMULATION 45 AND SUMMED EARLY-MINUS-LATE ACCUMULATION 47 115
ROTATE THE IN-PHASE AND QUADRATURE PROMPT ACCUMULATIONS 45 AND EARLY-MINUS-LATE ACCUMULATIONS 47 117
YES
MORE CHANNELS? 119
NO
RESET FOR ANOTHER ACCUMULATION INTERVAL 121
TIME PERIOD ELAPSED? 123
NO
SLEEP 125
YES
A

FIG. 4B

81 83 85

Table Element	Code Time Offset	Bit Sequence of L Code Chips (first is left-most, last is right-most)
$x(1)$	Δt_{0kmin}	0 0 … 0 0 0 0 0
$x(2)$	Δt_{0kmin}	0 0 … 0 0 0 0 1
$x(3)$	Δt_{0kmin}	0 0 … 0 0 0 1 0
$x(4)$	Δt_{0kmin}	0 0 … 0 0 0 1 1
⋮	⋮	⋮ ⋮
$x(2^L)$	Δt_{0kmin}	1 1 … 1 1 1 1 1
$x(2^L+1)$	$\Delta t_{0(kmin+1)}$	0 0 … 0 0 0 0 0
$x(2^L+2)$	$\Delta t_{0(kmin+1)}$	0 0 … 0 0 0 1 0
⋮	⋮	⋮ ⋮
$x(2^L \times k_{tot})$	Δt_{0kmax}	1 1 … 1 1 1 1 1

FIG. 5

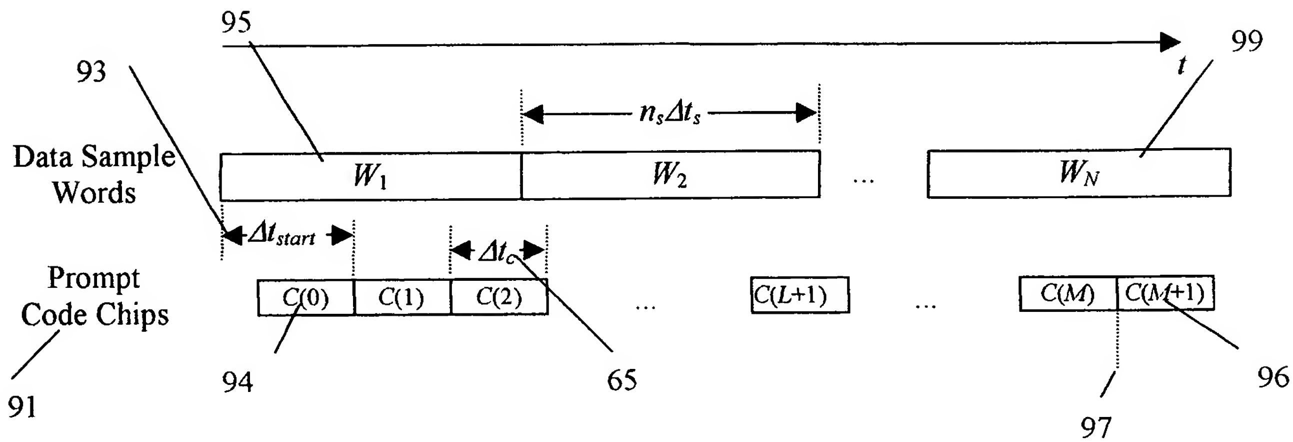

95
99
93
t
$n_s \Delta t_s$
Data Sample Words
W_1
W_2
...
W_N
Δt_{start}
Δt_c
Prompt Code Chips
C(0)
C(1)
C(2)
...
C(L+1)
...
C(M)
C(M+1)
91
94
65
96
97

FIG. 6

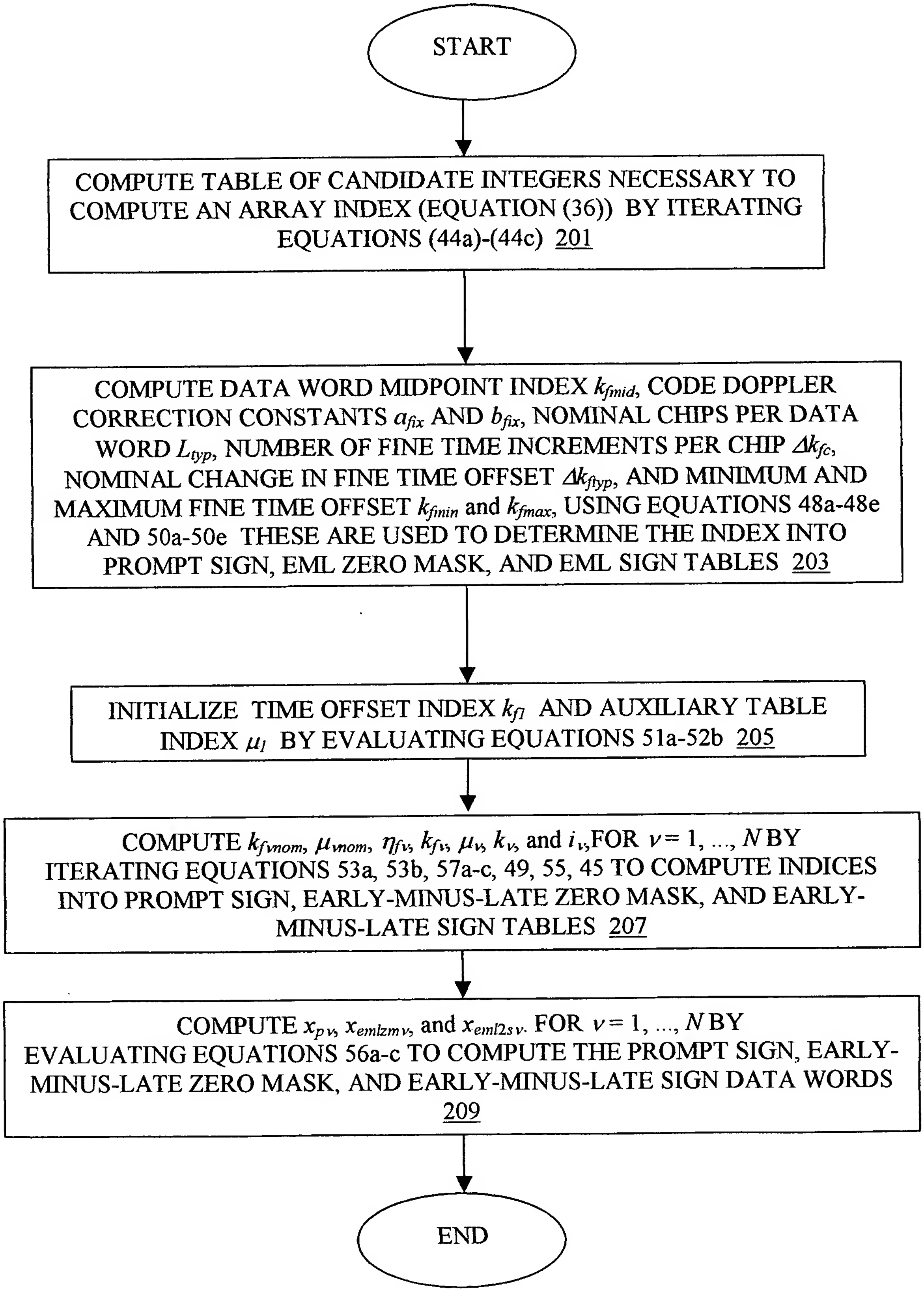

START
COMPUTE TABLE OF CANDIDATE INTEGERS NECESSARY TO COMPUTE AN ARRAY INDEX (EQUATION (36)) BY ITERATING EQUATIONS (44a)-(44c) 201
COMPUTE DATA WORD MIDPOINT INDEX k_{fmid}, CODE DOPPLER CORRECTION CONSTANTS a_{fix} AND b_{fix}, NOMINAL CHIPS PER DATA WORD L_{typ}, NUMBER OF FINE TIME INCREMENTS PER CHIP Δk_{fc}, NOMINAL CHANGE IN FINE TIME OFFSET Δk_{ftyp}, AND MINIMUM AND MAXIMUM FINE TIME OFFSET k_{fmin} and k_{fmax}, USING EQUATIONS 48a-48e AND 50a-50e THESE ARE USED TO DETERMINE THE INDEX INTO PROMPT SIGN, EML ZERO MASK, AND EML SIGN TABLES 203
INITIALIZE TIME OFFSET INDEX k_{f1} AND AUXILIARY TABLE INDEX μ_1 BY EVALUATING EQUATIONS 51a-52b 205
COMPUTE $k_{f\nu nom}$, $\mu_{\nu nom}$, $\eta_{f\nu}$, $k_{f\nu}$, μ_ν, k_ν, and i_ν,FOR $\nu = 1, \ldots, N$ BY ITERATING EQUATIONS 53a, 53b, 57a-c, 49, 55, 45 TO COMPUTE INDICES INTO PROMPT SIGN, EARLY-MINUS-LATE ZERO MASK, AND EARLY-MINUS-LATE SIGN TABLES 207
COMPUTE $x_{p\nu}$, $x_{emlzm\nu}$, and $x_{eml2s\nu}$. FOR $\nu = 1, \ldots, N$ BY EVALUATING EQUATIONS 56a-c TO COMPUTE THE PROMPT SIGN, EARLY-MINUS-LATE ZERO MASK, AND EARLY-MINUS-LATE SIGN DATA WORDS 209
END

FIG. 7

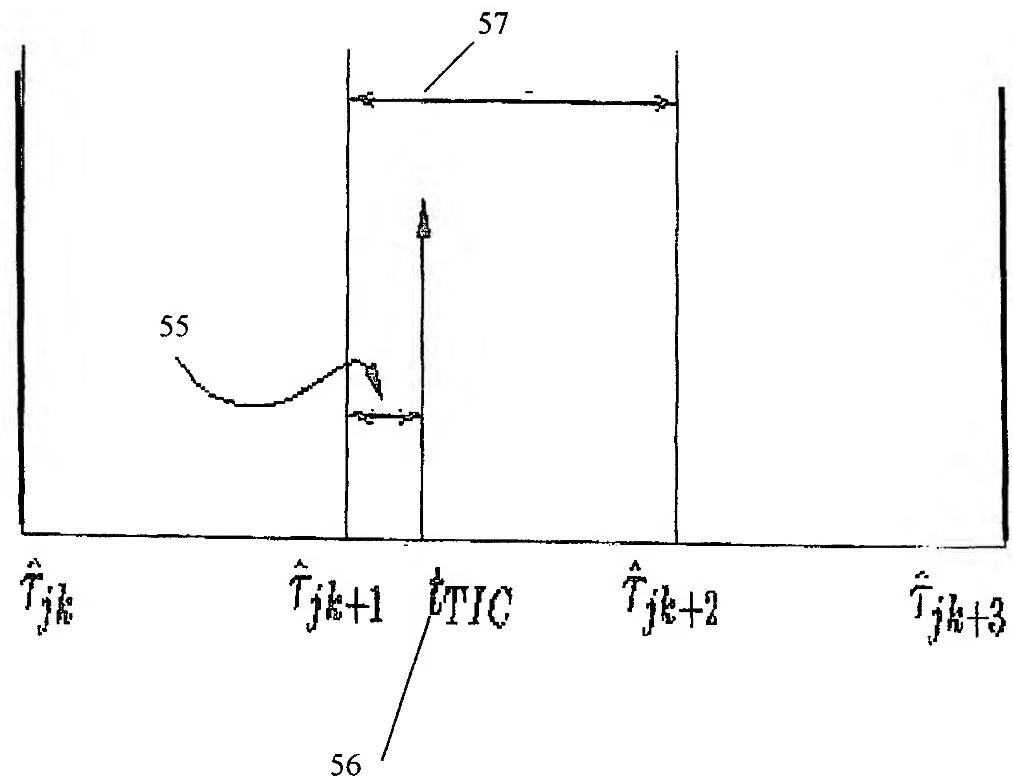
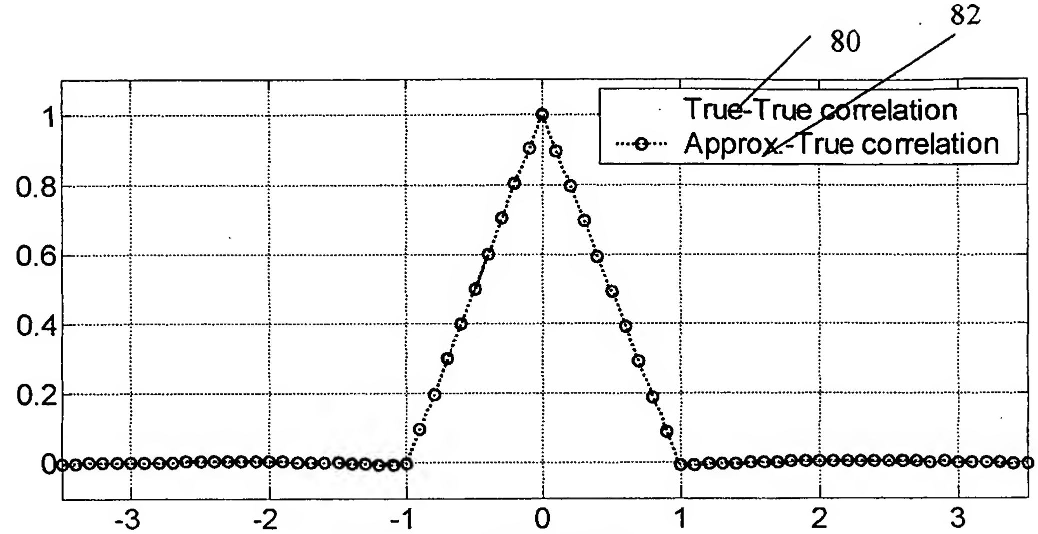
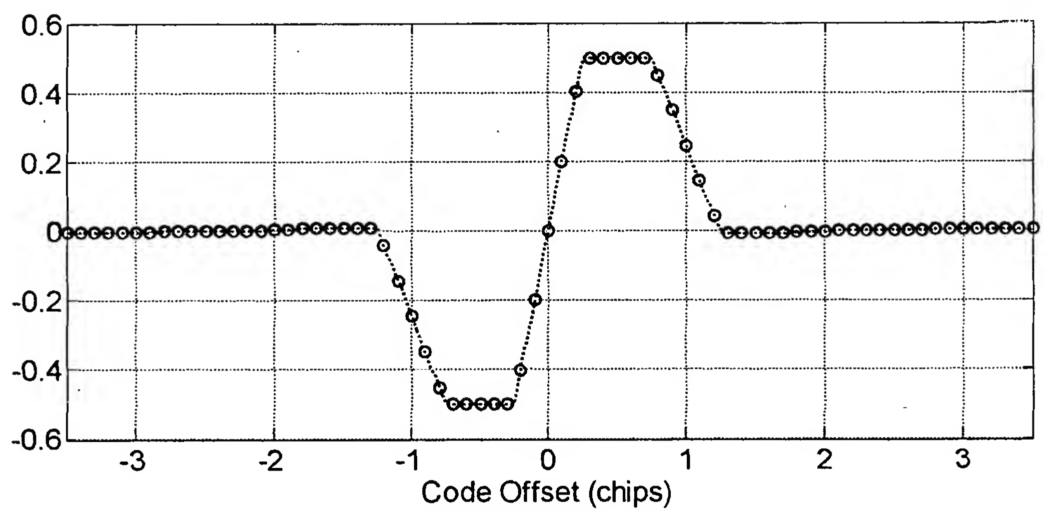


FIG. 8



80
82
True-True correlation
Approx.-True correlation
1
0.8
0.6
0.4
0.2
0
-3
-2
-1
0
1
2
3

FIG. 9A



0.6
0.4
0.2
0
-0.2
-0.4
-0.6
-3
-2
-1
0
1
2
3
Code Offset (chips)

FIG. 9B